Hedgerow object detection in very high-resolution satellite images using convolutional neural networks

Steve Ahlswede[a,*] **Sarah Asam**[b] **and Achim Röder**[a]
[a]University of Trier, Department of Environmental Remote Sensing and Geoinformatics, Trier, Germany
[b]German Aerospace Center, German Remote Sensing Data Center, Wessling, Germany

Abstract. Hedgerows are one of the few remaining natural landscape features within European agricultural areas. To facilitate hedgerow monitoring, cost-effective and accurate mapping of hedgerows across large spatial scales is required. Current methods used for automatic hedgerow detection are overly complicated and generalize poorly to larger areas. We examine the application of transfer learning using two neural networks (Mask R-CNN and DeepLab v3+) for hedgerow mapping in south-eastern Germany using IKONOS imagery. We demonstrate the potential of such networks for hedgerow monitoring by investigating performances across varying input image bands, seasonal imagery, and image augmentation strategies. Both networks successfully detected hedgerows across a large spatial scale (562 km^2), with DeepLab v3+ (75% $F1$-score) outperforming Mask R-CNN. Differences between band combinations were minimal, implying hedgerow detection could be achieved using RGB sensors. Results suggested that using all available training images across seasons is preferred and should have the same model generalizing effects as data augmentation. Experiments with varying data augmentations found augmentations effecting object geometries to greatly increase performance for both networks while results using augmentations modifying pixel spectral values showed concerning effects. Overall, our study finds that transfer learning in neural networks offers a simplified approach that outperforms previously established methods. © *2021 Society of Photo-Optical Instrumentation Engineers (SPIE)* [DOI: 10.1117/1.JRS.15.018501]

Keywords: deep learning; image segmentation; data augmentation; hedgerow mapping; Mask R-CNN; DeepLab v3+.

Paper 200608 received Aug. 16, 2020; accepted for publication Dec. 3, 2020; published online Jan. 5, 2021.

1 Introduction

Given the increasing human population, growing pressure is being put on natural resources. In particular, the demand for food, fiber, and energy products has led to extensive conversion of natural habitats to agricultural land as well as an intensified use of existing agricultural land. Thus agriculture has become one of the major drivers of land cover change and habitat loss globally.[1–5] The intensification of agriculture commonly results in the homogenization of agricultural fields, which, while beneficial from a production standpoint, has led to the loss of biodiversity and ecosystem services in agricultural areas.[6–8]

Hedgerows are linear landscape features comprised of shrubs and trees, traditionally established along the edges of agricultural fields as a delineation of properties or natural fencing systems.[9] They are cosmopolitan landscape features, which can be found throughout Europe.[10] Although they account for only a small fraction of agricultural land use areas, they provide numerous benefits within agricultural landscapes pertaining to biodiversity, hydrology, as well as soil conditions.[10–18]

*Address all correspondence to Steve Ahlswede, ahlswedes@gmail.com

1931-3195/2021/$28.00 © 2021 SPIE

Despite the positive benefits gained from hedgerows, their abundances have declined in Europe, due mainly to the intensification of agriculture.[13,14,18] In several European countries, financial incentives have been introduced to encourage sustainable farming practices such as hedgerow maintenance,[14,15] whereas other countries have explicitly included hedgerow protection into their Good Agricultural and Environmental Conditions.[19,20]

Hedgerows have historically been mapped through either field surveys or manual digitization of aerial imagery (S. Boell, Bayerisches Landesamt fuer Umwelt, LfU, pers. comment). *In situ* mapping leads to very precise results as well as the retrieval of auxiliary characteristics of the hedgerows (e.g., species composition and hedgerow height), but it is time-consuming and costly.[21] In the German state of Bavaria, hedgerows are often mapped by manually digitizing hedgerow boundaries using aerial imagery. However, this too becomes time-consuming and labor intensive when performed over large areas, leading to less frequent monitoring intervals, and reducing the ability to monitor trends.[11,22] Thus for regional monitoring of hedgerows, an automated and cost-effective approach would be preferred.

To date, automated hedgerow mapping from aerial or satellite imagery has focused on random forest or support vector machine methods using object-based image analysis (OBIA).[20,22–26] OBIA allows for the incorporation of object features such as size, shape, or context with regards to neighboring objects. Although increased inclusion of features have shown to improve hedgerow detections,[22,23,25,26] the lack of transferability of features across study sites limits the capacity of OBIA approaches.[11,22,25] Furthermore, manually designed features may be overspecified, incomplete, and time-consuming in terms of design and validation.[27] The use of manually engineered features is thus one of the main drawbacks to an OBIA approach for hedgerow mapping.[11,22,25,27] To facilitate the ability of non-experts in feature engineering and remote sensing to perform automated hedgerow mapping, the overhead of feature engineering must be reduced.

Convolutional neural networks (CNN)[28] are an alternative to OBIA, but have not yet been applied to hedgerow detection. CNNs applied to satellite and aerial imagery have shown improved results over other popular machine learning approaches.[29–33] They are particularly effective in classifying data with differing statistical distributions and complex decision boundaries without the use of manually engineered features.[34] Instead, CNNs autonomously generate the most relevant features for a given classification task through an iterative learning process.[27] These learned features can be applied to numerous computer vision tasks, including the localization and semantic labeling of objects within an image.[35] For the localization and labeling of image objects, two approaches can be employed, namely, semantic segmentation and instance segmentation.[36] Fully convolutional network (FCN)[37] is one possible approach to semantic segmentation, whereas networks that generate object proposals, such as regional CNNs (R-CNNs),[38] are often used for instance segmentation.

CNNs do reduce the difficulties associated with feature engineering and selection. However, the design, tuning of hyper-parameters, and proper training of CNN architectures require a great deal of expert knowledge as well as large amounts of training data, thus constraining their widespread application. In order to operationalize CNNs, the use of transfer learning (pretrained) and open source networks offer significant benefits. At the time of writing, DeepLab v3+ [39] (semantic segmentation) and Mask R-CNN[35] (instance segmentation) are both state-of-the-art with regards to their respective segmentation tasks and are openly available through public repositories.

The use of pretrained networks for remote sensing data does place restrictions on the input data, as pretrained networks are typically trained from large datasets of images containing three bands, and subsequent training must match this. Thus only three remote sensing bands can be utilized. Another limiting factor with regards to CNNs for remote sensing tasks is the often limited size of remote sensing training data, which can lead to overfitting of the network. A common technique used to offset small dataset sizes is the use of data augmentation. Here the original images are modified (e.g., rotated and scaled) in order to increase the size of the dataset[40] and avoid overfitting.[41] However, appropriate data augmentation strategies vary between datasets as well as prediction targets, as an inappropriate choice can result in negative effects on network predictions.[42,43]

Neither of the two networks examined here have been applied for the purpose of hedgerow detection, although studies involving alternative FCN and R-CNN architectures for vegetation classification have been carried out with success.[29,30] Zhao et al.[44] compared the performance of an FCN known as U-Net[45] and Mask R-CNN by performing tree canopy segmentation. Their results showed that Mask R-CNN outperformed the FCN on all measures of precision, with the downside being that Mask R-CNN required longer training time.

Although both networks have been shown to achieve near perfect accuracies on classification datasets such as COCO, this high level of performance does not always carry over when these networks are applied to novel datasets.[44] One reason is related to the quality of annotations used for network training, as annotation precision has been shown to influence network performance.[46] The creation of the COCO dataset included multiple quality checks to ensure precise object annotations.[47] However, monitoring agencies often will often lack the resources to invest in precise dataset annotations, resulting in roughly annotated datasets from which the network must learn (Fig. 11). Thus this research sets out to investigate the applicability of two high-performance networks for remote sensing object detection using real world datasets.

Given that OBIA approaches have been shown difficult to implement for non-experts with poor performance across varying study areas, this work systematically evaluates the potential of pretrained NNs to provide accurate and precise hedgerow detections in a practical manner. This is achieved by comparatively evaluating the performances between the state-of-the-art Mask R-CNN and DeepLab v3+ networks, as well as investigating the optimal data inputs for fine-tuning the networks. Specifically, we investigate the following aspects:

- the optimal three-band combination to be used with a pretrained network;
- the optimal seasonal imagery to utilize as input;
- the optimal data augmentation strategy for vegetation detection;
- assess and compare the performances between Mask R-CNN and DeepLab v3+ for hedgerow detection.

Using the best practice guided by the above investigations, we produce a hedgerow map across a large regional scale, demonstrating the scale at which neural networks are capable of making hedgerow detections. We further demonstrate the applicability of IKONOS, or any other 1-m resolution data, for regional hedgerow detection, as previous methods applied at this scale relied on sub-1-m resolution imagery.

The deep learning methods applied here are novel within the domain of hedgerow mapping and are relatively unexplored within the sphere of vegetation mapping from satellite imagery. As such, the results of this work should act as a guide for those wishing to perform landscape feature detection with pretrained NNs. Additionally, by applying and assessing well established NN architectures, we gain insights into which architecture features are beneficial to the task of hedgerow detection, thus providing some prior knowledge for future research that may aim to design a custom network specifically for hedgerow detection.

2 Study Area

The study area is located in Freyung-Grafenau, an administrative district located within the Eastern region of the federal state of Bavaria, Germany (Fig. 1). The district covers an area of 984 km^2, with a population of about 82,445, making it one of the most sparsely populated regions within Bavaria.[48] This is partly due to a large portion of the district overlapping with the Bavarian Forest National Park. The district is bordered by the Czech Republic to the northeast and Austria in the southeast.

Agricultural land-use covers a total of 30,263 hectares (ha), comprising 30.8% of the land-use within the district. Of this, only 5542 ha are used for arable farming, with the main crops being cereals such as spring barley, oats, as well as corn. The focus of agricultural land use is rather on animal husbandry of cow, cattle, and sheep.[48]

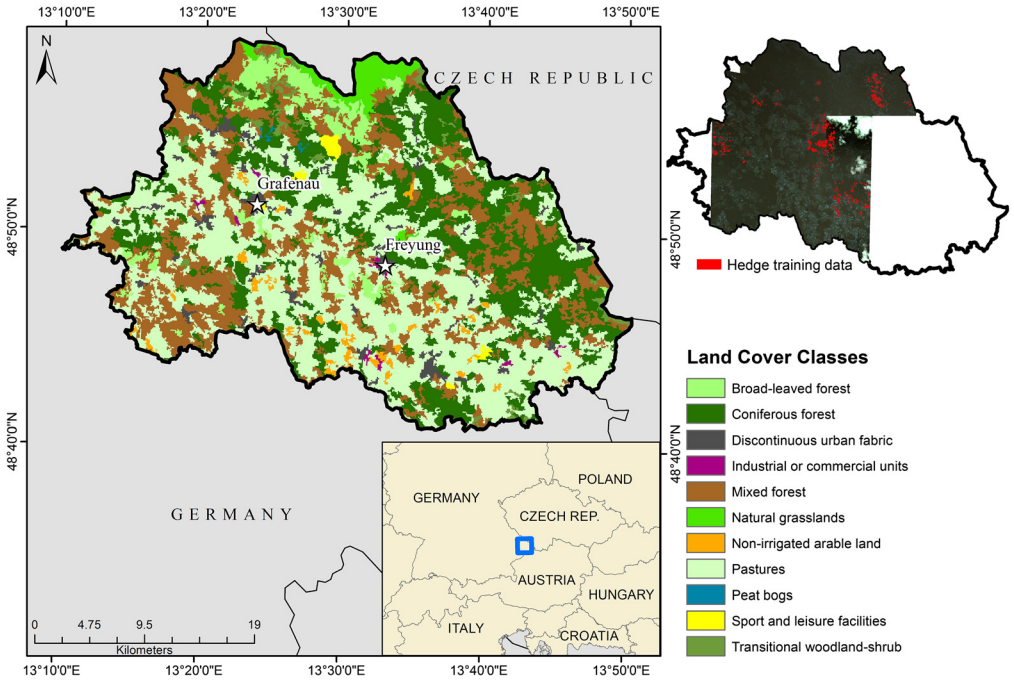

Fig. 1 Study area of Freyung-Grafenau as well as the CORINE land cover for the area. Top-right shows the coverage of available training data as well as the IKONOS imagery available to for the study.

3 Data

3.1 *Satellite Data*

IKONOS data were utilized due to the long historic range of IKONOS data compared to other alternative high-resolution datasets, making it an excellent candidate for the purpose of long-term hedgerow monitoring.

Five scenes from the IKONOS sensor were utilized. Two images were acquired on May 14, 2008, whereas the other three were acquired on October 8, 10, 13, 2007. IKONOS acquires panchromatic images with roughly 1-m resolution, as well as multispectral images in the blue (445 to 516 nm), green (506 to 595 nm), red (632 to 698 nm), and near-infrared (NIR) (757 to 853 nm) spectra at roughly 4-m resolution.[49] A pansharpening was performed using the Gram–Schmidt algorithm[50] in ArcMap (version 10.5.1, ESRI).

3.2 *Field Data*

Ground truth hedgerow polygons were obtained from the LfU. The dataset contained hedgerows that were manually digitized based on aerial imagery after determining hedgerow coordinates from field surveys carried out over the time period of 1984–2019. Given that some hedgerow polygons had been digitized more than 20 years before the IKONOS images were taken, some hedgerows had been lost or degraded. Thus manual inspection was carried out to ensure that hedgerow polygons indeed outlined woody vegetation within the IKONOS scenes. LfU defined hedgerows as naturally occurring linear structures comprised of a mixture of bush and tree species, being a minimum of two years old. As such, not all linear woody vegetation structures in images were labeled as hedgerows.

Given that the dataset spans over a period of 20 years, hedgerow polygons were digitized by numerous employees of the LfU. Differences in digitization quality thus exist across the dataset. As such, numerous areas have hedgerow objects whose boundaries were not digitized with high precision, causing object edges to be left out (Fig. 11).

3.3 *Data Preprocessing*

3.3.1 *IKONOS imagery*

Full IKONOS scenes were split into image tiles of 320×320 pixels. Tiles that contained ground truth polygons were used as training and validation inputs for the two NN, whereas tiles without hedgerow ground truth polygons were discarded. The tiling of images is necessary as larger sized images exhaust GPU memory resources.

Hedgerows appearing at the edges of image tiles lacked the contextual information that would be available if the object was located within the center of a tile. In order to reduce this effect, extracted image tiles had 64 pixels of overlap with neighboring image tiles.[30,51] This further served to increase the size of the training dataset.

3.3.2 *Field data*

Annotated images are required when training Mask R-CNN or DeepLab v3+ for the network to learn, which pixels belong to the target class(es). Thus a binary mask was required for each 320×320 IKONOS image tile, where pixels with a value of 1 belonged to the hedgerow class, and pixels with a value of 0 belonged to the background class. Hedgerow polygon shapes were thus converted to mask images using ENVI (version 5.1, Exelis Visual Information Systems).

In the case of DeepLab v3+, single annotation images containing multiple individual hedgerows could be used. However, as Mask R-CNN performs instance segmentation rather than semantic segmentation, annotation images could only delineate a single individual hedgerow. Thus in cases where multiple hedgerows appeared in a single-image tile, separate annotation images were required for each hedgerow.

Due to the tiling of images, small portions of hedgerows were often cut off along image edges. Small mask regions of <200 pixels were thus removed from mask images, as to provide only properly shaped hedgerows for network training.[44] Mask images void of hedgerows were then removed, leading to a dataset containing a total of 684 tile images. The dataset was then split into training and validation sets with a 75%/25% split, respectively, resulting in 513 training and 171 validation tiles (Fig. 12).

4 Methods

4.1 *Mask R-CNN*

Mask R-CNN is the most recent iteration of the R-CNN family.[35,38,52–54] The network can be broken into four main components: a CNN, which performs feature extraction; a feature pyramid network (FPN);[55] a region proposal network (RPN); and the network "heads" producing the final network output.[35] Numerous CNN architectures can be used within Mask R-CNN. Here we chose to use the ResNet-101[56] architecture as the authors of Mask R-CNN have found it to outperform other alternatives.[35]

A CNN extracts features by convolving the image with learned convolutional filters.[56] A set of convolutional filters are thus known as a convolutional layer. Early CNN layers detect simple features such as edges or color blobs. As the feature maps are passed deeper through the network, filters combine lower features to form increasingly complex features.[57] Pooling is typically performed in order to encode translation invariant features while also reducing the resolution of feature maps and thus computation time.[28] Thus the final feature layers of a CNN contain highly semantic features, but given the loss of resolution, these layers have a reduced capacity to accurately localize objects within the original image. The FPN of Mask R-CNN attempts to restore the spatial resolution of the final CNN layer by sequentially upsampling the feature maps.[55]

Feature maps from the FPN are passed to the RPN subnetwork of Mask R-CNN. Here feature maps are scanned using windows known as "anchors," which slide across feature maps looking for possible objects. A total of 15 different sized anchors are formed based on five scale and three

aspect ratio hyperparameters. Larger sized anchors are applied to FPN layers with coarser resolutions while smaller anchors are applied to the fine-scale resolution FPN layers. This greatly aids in the detection of objects across multiple scales as the scale of features are able to better match the scale of the target object.[35] Probability scores for two general classes of "background" and "object" are calculated for each anchor, allowing anchors with low object probabilities to be filtered out. The final sets of anchors are passed on to the network heads for classification into more specific object classes.

Three network heads are used in Mask R-CNN. The first performs a regression on the bounding box coordinates in order to improve the accuracy of the detected object's bounding box size and location. The second head performs a classification where the class with the highest probability becomes the object's label. The final head produces a pixel-wise mask of the object within the area of the predicted bounding box.[35]

4.2 *DeepLab v3+*

DeepLab v3+ is an FCN network that performs semantic segmentation using an encoder–decoder structure.[39] The encoder module of DeepLab v3+ uses a modified version of a CNN known as Xception.[58] Atrous convolution is used instead of performing the final pooling operation in Xception. As atrous convolutions space out the convolutional filter such that the filter window has an expanded field of view without increasing the filter dimensions,[39] their use preserves spatial detail in the feature maps and leads to less downsampling compared to Mask R-CNN. Thereafter, a series of atrous convolutional layers with successively increased rates of spacing are used to form an atrous spatial pyramid pooling (ASPP) layer. This allows for the capture of object and contextual features at multiple spatial scales. Outputs from each of the atrous convolutions of the ASPP are merged to form the final feature maps of the encoder module. DeepLab v3+ uses rates of 6, 12, and 18 for the ASPP.[39]

Given the feature maps are 16 times smaller than the original image size, the decoder module restores the feature maps to the original resolution. In order to enhance the spatial precision of mask predictions, this upsampling is broken into multiple stages in order to restore the feature maps to the original image dimensions.[39] Feature scores for each pixel are then used to estimate the probability of a given pixel belonging to each of the target classes.[59]

4.3 *Hyperparameter Configurations*

A single Geforce RTX 2080 Ti GPU with 11 GB of memory was used, leading to batch sizes (number of images used per-training step) of six. When a network has been trained on all images of the dataset once, it is known as an epoch. For each of the experiments (see Sec. 4.4), the networks were trained for a total of 100 epochs using an initial learning rate of 0.001. Learning rates used a "poly" learning rate policy.[60] Thus after each training step the initial learning rate was multiplied by Eq. (1), where step refers to a single batch of images, and stepmax refers to the total number of training steps:

$$[1 - (\text{step} \div \text{stepmax})]^{0.9}. \tag{1}$$

Due to the thin nature of hedgerows, images were dominated by background pixels. Such class imbalances can bias the loss function, leading to poor classification results.[61] In such cases, an accuracy measure could achieve high-accuracy rates by labeling all pixels as background. To avoid this, class specific losses were weighted in favor of the hedgerow class in order to emphasize the correct labeling of this class. For DeepLab v3+, a weighting value of 350 was chosen for hedgerows, whereas background was left at 1. Values above 350 were found to cause problems with the training, resulting in loss values growing toward infinity. Class imbalance was less of a problem for Mask R-CNN given that images are first segmented into smaller regions (RoIs) where the target object is dominant. As such, no class weights were applied for Mask R-CNN.

The setting of appropriately sized anchors is important in accurate object detection within the Mask R-CNN network, as these anchors perform the initial region proposals. Thus if anchors sizes are set inappropriately the network will struggle to accurately locate objects.[62] Anchor aspect ratios where length was much greater than the width for both horizontal and vertical cases were found to greatly improve performance. As hedgerows may also run diagonally across an image, a square shaped aspect ratio was also used, resulting in aspect ratios of 0.2, 1.2, and 5. Additionally, the "scale" parameter for the anchors needs to be adjusted so that the size of anchors loosely resembles that of the ground truth boxes. The anchor scale parameters were set to values of 30, 60, 100, 150, and 200. Using 150 and 200 resulted in elongated anchors becoming larger than the image itself (Fig. 2) but was necessary in order to increase the size of the square shaped anchor that was meant to capture long diagonal hedgerows running across the image.

Pretrained network weights were used for both DeepLab v3+ and Mask R-CNN. The utilization of pretrained weights is a practice known as transfer learning, which has been shown to greatly improve results across various image analysis tasks.[27,53,63] Networks first train on a large secondary dataset in order to learn low-level features that are universally useful in object detection (e.g., edges and color blobs).[57] The learned features are then subsequently fine-tuned using the target dataset. Since low-level features are generally useful for most image analysis tasks, these layers were frozen during fine-tuning while upper layers of the network were fine-tuned to adjust to the target classification.[64] Weights utilized as a starting point for both networks were derived from pretraining on the public image dataset COCO.[47] Given that objects in the COCO dataset are often amid clutter, partially occluded, or residing in the background, networks trained on this dataset place increased importance on learning context based features, leading to increased performance in the detection of objects within natural contexts.[47] This should thus help with the detection of hedgerows as spectral information alone can lead to misdetections with other vegetation, whereas contextual information, such as placement within agricultural, fields is important.

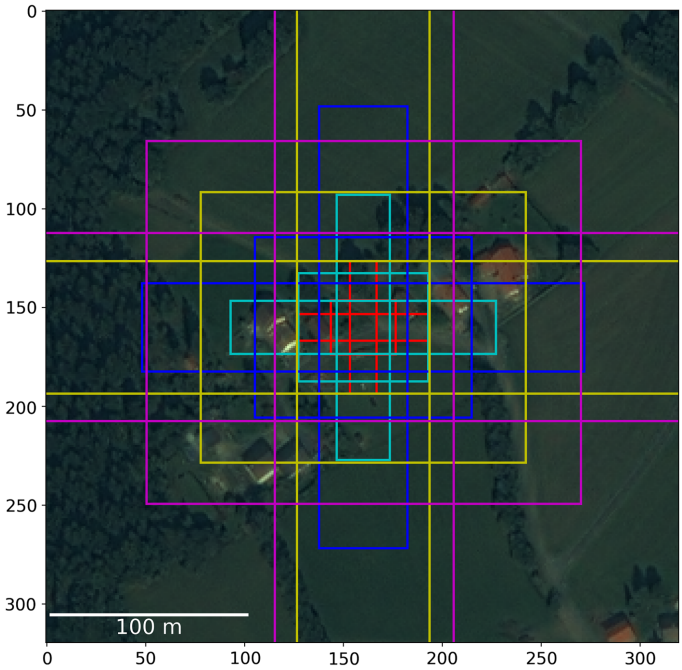

Fig. 2 Range of anchors produced by the region proposal network. Each color represents a different scale.

4.4 *Experimental Setup*

4.4.1 *Optimal three-band combination*

The use of pretrained NN allows for a simplification of the training process, as such networks have learned robust low-level features that can be transferred to numerous object detection tasks.[63] However, pretrained networks are predominantly trained on three band (RGB) images. Thus fine-tuning using the target dataset must also be done using three band inputs. Although pretrained networks have been fit to RGB data, previous works on hedgerow detection have found the NIR band as one of the most significant features for classifiers trained to detect hedgerows.[20,22,23] Thus in order to investigate whether hedgerow detection can be achieved using a more limited set of bands (e.g., RGB), we investigate whether the inclusion of the NIR band provides improvements when fine-tuning Mask R-CNN and DeepLab v3+. As such, all four possible band combinations for IKONOS were tested.

4.4.2 *Optimal seasonal imagery input*

The IKONOS images available for analysis were obtained from two different seasons, with half of the dataset being from October, and the other half from May. A shift in the phenological phase between the two time periods leads to different spectral signals between hedgerow objects. It was thus investigated whether the use of data from different seasons influences network performance. Three datasets were tested: one containing only October images, one containing only May images, and one containing a mixture of the two. In order to control for the effect of training dataset size, all three datasets were limited to a size of 247 images. Augmentations (see Sec. 4.4.3) were applied to each dataset, leading to total dataset sizes of 5434 training images for each of the datasets. Validation datasets were also standardized between the three seasons, resulting in 87 validation images for each season.

4.4.3 *Optimal data augmentation strategy*

The consensus within the literature is that data augmentation leads to improved model results.[40,41,65,66] Augmentation allows for robust recognition of objects by exposing the network to variations of object features.[28] For example, by applying random noise to image pixel values, the network learns to rely less on the typical spectral signatures and identifies objects based on other defining features. This should allow the network to reduce false detections of spectrally similar wooded vegetation (e.g., forest edges) by placing increased emphasis on other defining features (e.g., contextual). Data augmentation strategies for remote sensing images typically rely on simple geometric augmentations such as rotation or translation, with some researchers applying spectral augmentations (e.g., adding noise).[67,68] However, no work has quantitatively examined the effects of these and other possible augmentations regarding the detection of vegetation.

Ten data augmentation strategies were applied, and the resulting model performances were measured in order to quantify the effects of each data augmentation type. Augmentations that mimic typical variations in the target object as well as in satellite imagery were chosen. Geometric augmentations such as scaling and rotation could naturally occur due to differences in flight altitude and flight path heading, respectively. Spectral augmentations could be caused by differing atmospheric effects, differences in illumination conditions, as well as detector defects leading to darker images or image noise.[69,70] All augmentations applied in this study are summarized in Table 1, including the shorthand names used to refer to each augmentation in this paper. Figure 3 shows an example of how an image changes under each augmentation. Augmentations were applied using the Python package imgaug (version 0.3.0).

Given that increases in dataset size lead to benefits in model performance, augmentations were examined individually rather than in a stepwise additive fashion to avoid the confounding effect of increasing dataset size. Thus each augmentation dataset contains 1026 images.

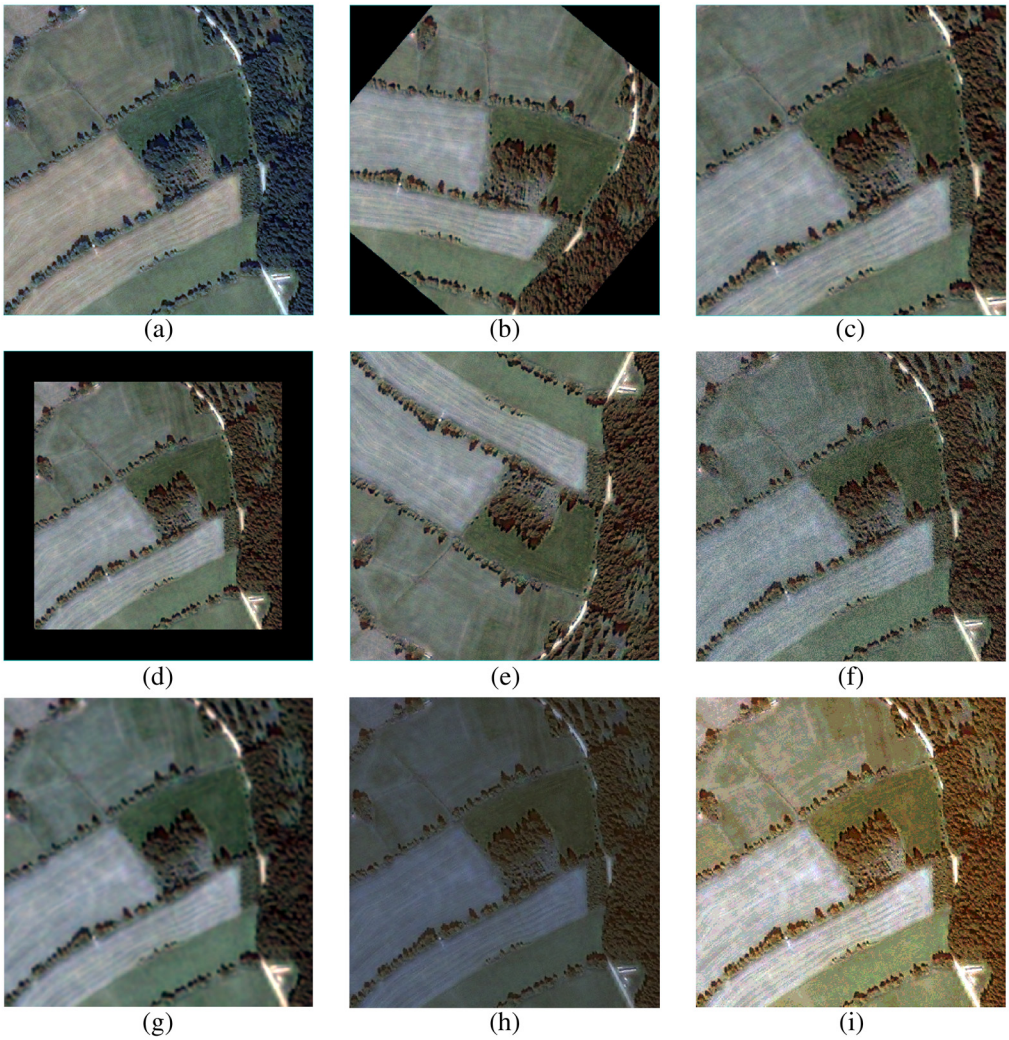

Fig. 3 An unaugmented image (a) and the image with augmentations applied. Augmentations involved (b) rotation, (c) scaling up and (d) down, (e) flipping the image, (f) adding noise, (g) applying Gaussian blur, (h) uniformly decreasing or increasing image pixels, and (i) applying log contrast.

In the case of rotated and scaled images, hedgerows at the edges of the image were sometimes cropped out by these augmentations, leading to some images being void of hedgerow objects and thus being discarded. The rotation 45 and scale datasets thus contained 989 and 946 images, respectively.

Augmentations were also applied together, forming two final augmentation datasets. One dataset contained only geometric augmentations (G), applying either one, two, or three randomly selected (without replacement) geometric augmentations per image. The other used both geometric and spectral augmentations (GS), applying one, two, or three randomly selected geometric augmentations and either zero, one, or two randomly selected spectral augmentations per image. For the final augmentation datasets, a larger range of possible rotations was used than those mentioned in Table 1, as testing all rotation angles individually was too time-consuming. Here rotations ranged from 40 deg to 320 deg with steps of 40 deg in between. Final augmented datasets for G and GS contained 12,655 images and 12,499, respectively. The difference in the size of the dataset was caused by the random application of augmentations, leading to differences in the number of images containing hedgerow objects.

Table 1 Shorthand names used for each of the different augmentations used in this study. Parameter settings as well as a description of how the image changes under each augmentation are provided.

Name	Parameter setting	Description
Add	Range: −80 to 80	Image pixel values are uniformly increased or decreased by a value randomly selected within the range parameter
Add per channel	Range: −80 to 80	Image pixel values are uniformly increased or decreased across each band. For each band, a new value is randomly selected from within the range parameter
Noise	Range: −60 to 60	Image pixel values are uniformly increased or decreased on a per-pixel basis. For each pixel, a new value is randomly selected from within the range parameter
Blur	Sigma: 0.75	Gaussian blurring is applied to each pixel with a sigma of 0.75
Contrast	Gain: 0.6 to 1.4	Image contrast is increased uniformly by a value randomly selected from within the range of the gain parameter
Contrast per channel	Gain: 0.6 to 1.4	Image contrast is increased on a per-channel basis. For each channel, a new value is randomly selected from within the range of the gain parameter
Flip	None	Images are either flipped along the horizontal or vertical axis
Rotate 45	None	Images are rotated by a degree of 45
Rotate 90	None	Images are rotated by a degree of 90
Scale	±20%	Images are either scaled up to 120% of the original image size or down to 80%

4.5 *Performance Metrics*

Given that traditional remote sensing performance metrics (e.g., confusion matrix) can be biased when large class imbalances exist,[71] object detection tasks often utilize precision and recall measures, as well as the $F1$-score.[44,72,73] Precision [Eq. (2)] and recall [Eq. (3)] are calculated based on objects, and thus a threshold for the overlap between a ground truth mask and a predicted mask is required to distinguish positive from negative detections. Here three thresholds were employed; $>70\%$, $>50\%$, and $>30\%$ overlap. Thus if a predicted hedgerow mask overlapped with the ground truth mask by more than the given threshold, it would be considered a true positive (TP) detection. A false positive (FP) occurs when a predicted mask does not meet the requirements of a TP. Finally, a false negative (FN) occurs when a ground truth mask does not adequately overlap with any predicted masks. Here the $>70\%$ threshold is considered the most important indicator of network performance. However, the inclusion of lower thresholds offered an improved quantitative representation of the network performances:

$$\text{precision} = \text{TP}/(\text{TP} + \text{FP}), \tag{2}$$

$$\text{recall} = \text{TP}/(\text{TP} + \text{FN}). \tag{3}$$

The $F1$-score [Eq. (4)] is calculated using the above precision and recall values:

$$F1 - \text{score} = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}}. \tag{4}$$

Per-pixel accuracy was also used to measure performance. However, as background pixels were not of interest and would lead to inflated rates of accuracy, per-pixel accuracy was calculated by looking at only areas where either the ground truth or predicted masks occurred:

$$\text{per -pixel accuracy} = \text{TP}_{\text{pixel}}/(\text{TP}_{\text{pixel}} + \text{FP}_{\text{pixel}} + \text{FN}_{\text{pixel}}), \tag{5}$$

where TP_{pixel}, FP_{pixel}, and FN_{pixel} refer to the above described measures with respect to pixels.

4.6 *Python Implementation*

All data preprocessing tasks, network training, and analysis of results were carried out using Python (version 3.7.1). The code for DeepLab v3+ was written by Chen et al.[39] and can be found in an online repository provided by TensorFlow. In this study, the Matterport Inc. implementation of Mask R-CNN, which was released under a Massachusetts Institute of Technology license was utilized. As TensorFlow currently does not support the reading of .tif image files, all image tiles were converted to .png format.

5 Results

5.1 *Optimal Three Band Combination*

Different band combinations were tested in order to find the optimal three band combination for fine-tuning the pretrained networks. Table 2 shows the results from each combination for both NNs.

With Mask R-CNN, per-pixel accuracy and $F1$-score were highest using GRNIR images. Differences in precision given a >70% TP threshold were minimal between the different band combinations. RGB and GRNIR both achieved the highest precision, whereas recall rates were highest in the GRNIR dataset. At the >50% threshold, BGNIR had the highest precision, whereas the highest recall came from GRNIR. At the >30% threshold, RGB had both the highest precision and recall.

With DeepLab v3+, hedgerow per-pixel accuracy was the highest using RGB bands, whereas the highest $F1$-score was attained from the BGNIR combination. At the >70% threshold, GRNIR had the highest precision score, whereas the highest recall was BGNIR. At the >50% threshold, precision was highest for GRNIR, whereas recall was the highest for

Table 2 Results of four possible band combinations for IKONOS at three thresholds for TP overlap (70%, 50%, and 30%). Bands include red (R), green (G), blue (B), and near infrared (NIR). Results for the Mask R-CNN network are denoted with (M), and DeepLab v3+ with (D). The highest result from a column for each method is bolded.

Band combination and network	Per-pixel accuracy (%)	$F1$ Score (70%)	Precision (70%)	Recall (70%)	Precision (50%)	Recall (50%)	Precision (30%)	Recall (30%)
RGB (M)	31.9	46.9	**46.5**	47.3	58.3	59.2	**65.2**	**66.3**
GR + NIR (M)	**32.6**	**47.5**	**46.5**	**48.6**	57.1	**59.7**	61.6	64.4
BR + NIR (M)	30.6	45.4	46.4	44.5	56.2	53.9	62.4	61.0
BG + NIR (M)	31.7	44.5	45.7	43.4	**58.7**	55.8	64.1	59.9
RGB (D)	**30.5**	58.3	60.1	56.7	69.6	65.6	78.0	73.5
GR + NIR(D)	30.3	60.1	**60.7**	59.5	**71.5**	70.1	76.3	75.1
BR + NIR(D)	29.4	59.5	60.5	58.6	70.4	68.2	**79.9**	**77.3**
BG + NIR (D)	30.2	**60.6**	57.6	**63.9**	64.2	**71.2**	68.6	76.1

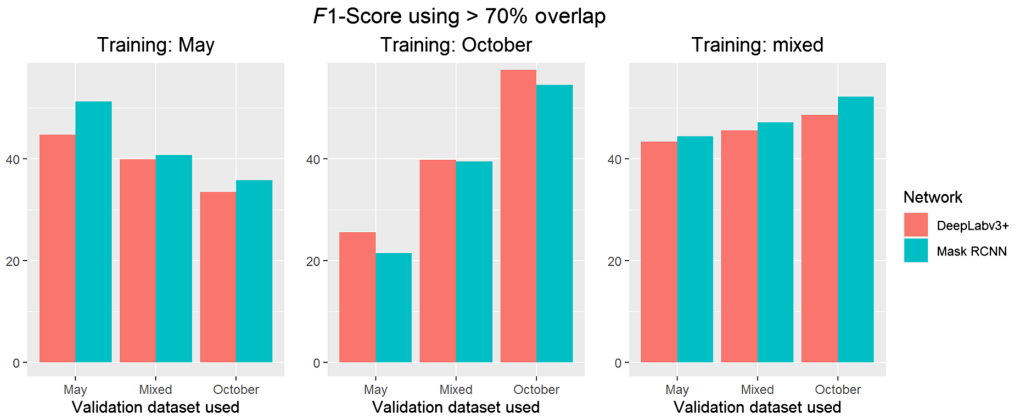

Fig. 4 *F*1-scores for both DeepLab v3+ and Mask R-CNN using three different seasonal datasets (May, October, and a mixture of the two) split into training and validation sets.

BGNIR. At the >30% threshold, precision was highest for BRNIR, whereas recall was highest for BRNIR.

GRNIR was deemed as the optimal band combination given high *F*1-score, precision, recall, and per-pixel accuracies at the >70% threshold between both networks. Thus all subsequent data analyses were performed using only the GRNIR bands.

5.2 *Optimal Seasonal Imagery Input*

Results showed that the highest per-pixel accuracies, as well as *F*1-score, precision, and recall (at >70% threshold) (Figs. 4, 13, 14, and 15) were achieved when using October images for both training and prediction. Per-pixel accuracy, *F*1-score, precision, and recall all decreased when using the mixed dataset at the prediction stage after training with images from October. Using images from May at the prediction stage after using October for training resulted in the lowest scores across all performance metrics.

Networks trained with the mixed dataset achieved the highest per-pixel accuracies, *F*1-score, recall, and precision when using images from October in the prediction stage. Using the mixed dataset in the prediction stage lowered performance across all metrics, whereas using May images for predictions resulted in the lowest performance across all metrics.

For networks trained with only images from May, results were highest using May images during the prediction stage. Using the mixed season images for predictions led to decreases in performance metrics scores, whereas using October images for prediction resulted in the lowest performance metrics scores.

5.3 *Optimal Data Augmentation Strategy*

5.3.1 *Individual augmentations*

Test results from all data augmentations for both Mask R-CNN and DeepLab v3+ are shown in Fig. 5. Here the non-augmented dataset ("none") is included as well for comparison.

With regards to Mask R-CNN, spectral augmentations all achieved improvements over none in the per-pixel accuracy of hedgerow predictions, with the addition of noise achieving the highest accuracy. With respect to none, per-pixel accuracies decreased across three of the four geometric augmentations (flip, rotate 45, and scale), with only rotate 90 leading to improvement. *F*1-score, precision, and recall rates were only examined at the >70% threshold for object detection. *F*1-score improved with respect to none in all but one ("add") of the spectral augmentations. All four geometric augmentations caused a decrease in the *F*1-score with respect to none. Precision improved with respect to none using three of the spectral augmentations (contrast, contrast per channel, and noise) and decreased using the two others (add and blur). Precision rates from geometric augmentations increased with respect to none using both flip and rotate 45

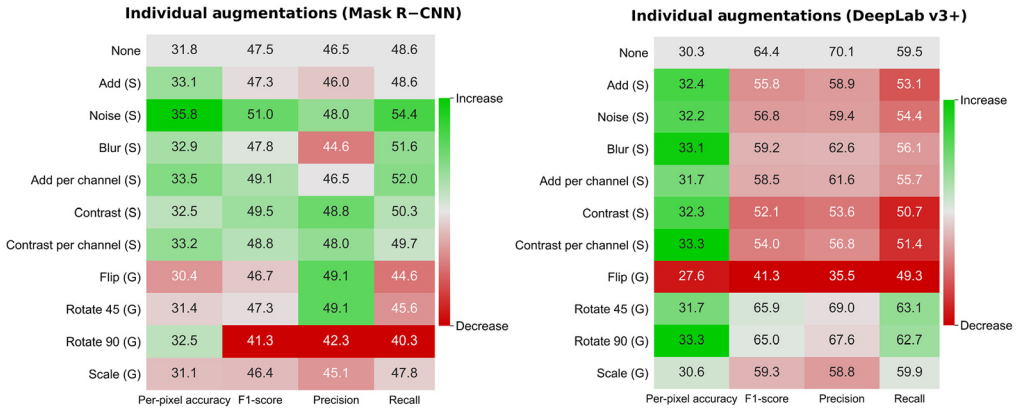

Fig. 5 Per-pixel accuracy, *F*1-score, precision, and recall for all augmented datasets. *F*1-score, precision, and recall are calculated using a >70% threshold to determine object detections. Increases and decreases are highlighted with respect to none (original dataset without augmentation). Spectral augmentations are labeled with (S), while geometrical augmentations are labeled with (G).

and decreased using scale and rotate 90. Recall improved with respect to none across all spectral augmentation datasets aside from add, which had the same recall rate as none. Recall rates decreased using all geometric augmentation datasets when compared to none.

With regards to the DeepLab v3+ network, per-pixel accuracies all improved compared to none aside from flip, with rotate 90, and contrast per channel having the highest per-pixel accuracy. *F*1-scores all decreased when using spectral augmentations and increased in two (rotate 45 and rotate 90) of the four geometric augmentations. Precision rates increased with respect to none using two of the six spectral augmentations (blur and add per channel) as well as with two of the four geometric augmentations (rotate 45 and rotate 90). Recall rates increased with respect to none for only three of the tested augmentations: rotate 45, rotate 90, and scale.

5.3.2 *Grouped augmentations*

Results from the fully augmented datasets are shown in Table 3. For Mask R-CNN, per-pixel accuracy was higher using *G*. At the >70% detection threshold, *F*1-score, recall, and precision were all higher using GS. At the >50% threshold, precision was higher using *G*, but recall was higher using GS. At the >30% threshold this pattern continued, with a highest precision using *G* but higher recall using GS. Looking at DeepLab v3+, all performance metrics were higher when using *G*.

Table 3 Results of using the fully augmented datasets at three thresholds for true positive overlap (70%, 50%, and 30%). Mask R-CNN is denoted by (M), whereas DeepLab v3+ is denoted with (D).

Augmentation	Per-pixel accuracy (%)	*F*1-score (70%)	Precision (70%)	Recall (70%)	Precision (50%)	Recall (50%)	Precision (30%)	Recall (30%)
None (M)	32.6	47.5	46.5	48.6	57.1	59.7	61.6	64.4
Geometric (M)	**40.6**	55.3	51.3	59.9	**60.3**	70.3	**64.8**	75.6
Geometric + spectral (M)	39.5	**58.6**	**53.6**	**64.6**	60.1	**72.4**	63.1	**76.1**
None (D)	30.3	60.1	60.7	59.5	71.5	70.1	76.3	75.1
Geometric (D)	**36.9**	**74.7**	**69.4**	**80.8**	**73.9**	**86.1**	**77.6**	**90.4**
Geometric + spectral (D)	36.4	73.5	68.1	79.9	70.9	83.1	73.8	86.5

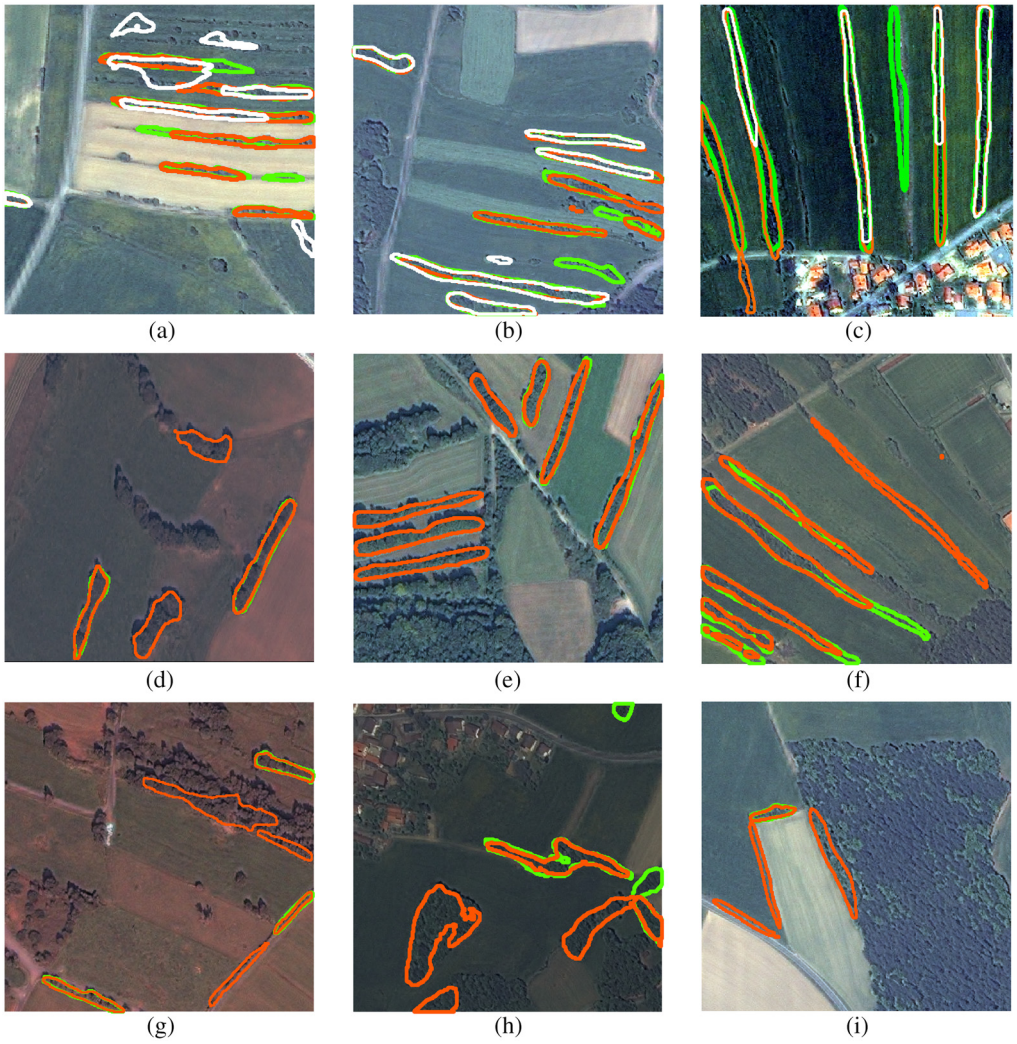

Fig. 6 Hedgerow predictions generated from Mask R-CNN using different augmentation strategies. Network predictions using the unaugmented dataset are shown in white, whereas predictions using geometric augmentations and geometric plus spectral augmentations are shown in green and orange, respectively. Tiles are each 320×320 pixels in size.

Figures 6 and 7 show images of some of the detections made by both Mask R-CNN and DeepLab v3+, respectively. In both figures, the top row highlights how both augmentation datasets improved mask predictions compared to training without augmentation, the middle row shows cases where the use of GS led to improvements over *G*, whereas the bottom row shows cases of the opposite.

With respect to Mask R-CNN, the application of either augmented dataset (*G* or GS) visually improved the quality of hedgerow masks. Hedgerow masks in areas of dense hedgerow occurrences were sometimes poorly masked without the added training gained from using augmented images [Fig. 6(a)]. Additionally, more hedgerows were detected using the augmented datasets [Figs. 6(b) and 6(c)], and the network was able to detect the full extent of longer hedgerows after augmentation had been used [Fig. 6(c)]. In the middle row of Fig. 6 is shown how GS led to increased detections of hedgerows, which were missed when using *G*. However, the increased TP detections that came from using GS led to numerous FP detections, as seen in the bottom row of Fig. 6. Forest edges that neighbored agricultural fields were more often misclassified as hedgerows when using GS [Figs. 6(h) and 6(i)] as well as non-linear clusters of woody vegetation within agricultural fields [Fig. 6(h)].

With respect to DeepLab v3+, augmentations again improved the quality of hedgerow masks when compared to the predictions made from the network trained on the unaugmented dataset

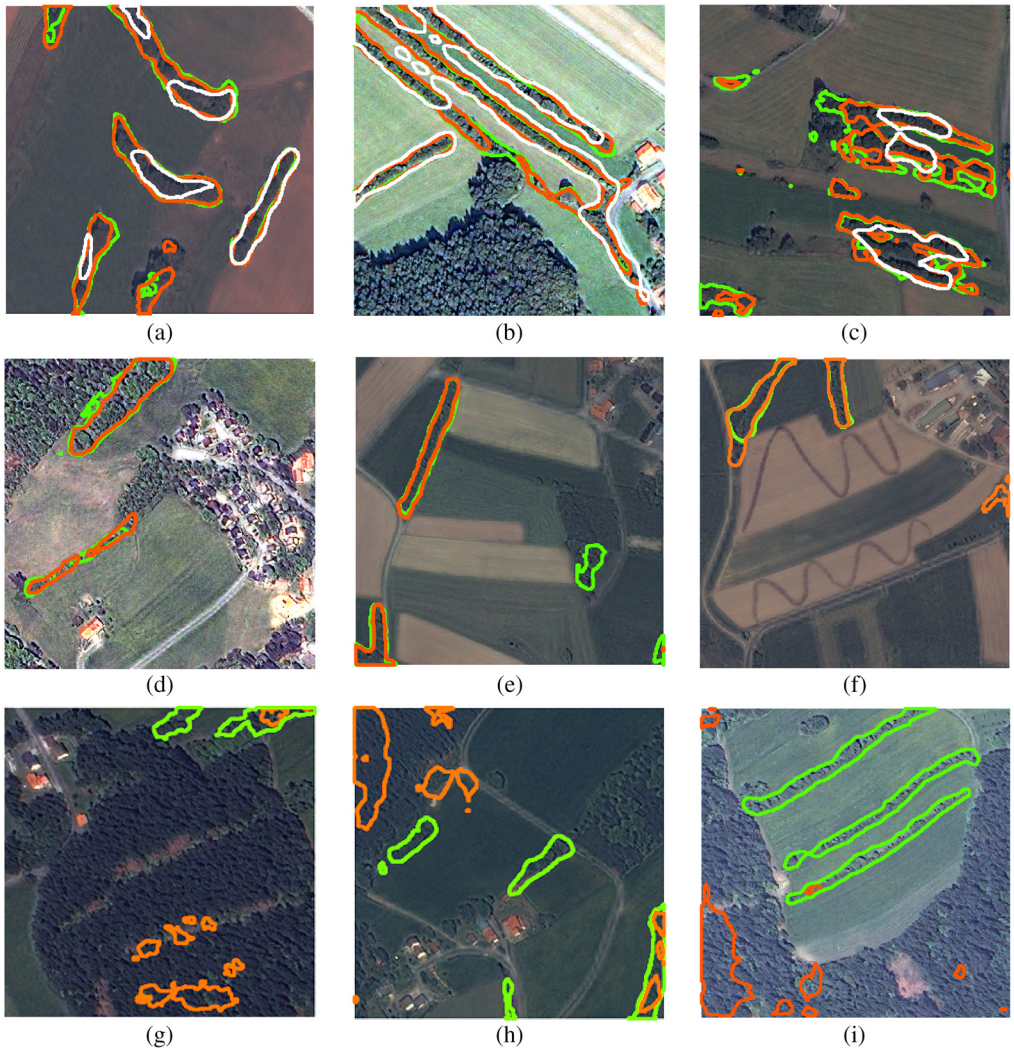

Fig. 7 Hedgerow predictions generated from DeepLab v3+ using different augmentation strategies. Network predictions using the unaugmented dataset are shown in white, while predictions using geometric augmentations and geometric plus spectral augmentations are shown in green and orange, respectively. Tiles are each 320×320 pixels in size.

(Fig. 7, upper row). Hedgerow mask boundaries were generally more accurate when training with the augmented datasets. Without augmentation, the separation of individual hedgerows was problematic, as predictions often included pixels from the field in between hedgerows, especially in areas of dense hedgerow occurrences [Figs. 7(b) and 7(c)]. However, in certain areas of dense occurrences, the separation of individual hedgerows remains an issue even after using augmented images [Fig. 7(c)]. The use of GS compared to G had some positive effects with regards to the accuracy of hedgerow prediction boundaries. As can be seen in row two of Figs. 7(d) and 7(f), there were cases where the boundaries of hedgerow masks are more precise when using GS compared to those made using G. However, this was not the case in all image predictions, as can be seen in the third row of Fig. 7. It illustrates how the use of spectral augmentations can lead to problematic FP detections within forested areas, whereas the hedgerows that were detected when using G are completely missed.

5.4 *Comparison Between Mask R-CNN and DeepLab v3+*

Based on the outputs of the final augmented datasets, a prediction map was produced from each of the fine-tuned models. For Mask R-CNN, the network trained on GS was used, whereas for DeepLab v3+ the network trained on G was used for the prediction map.

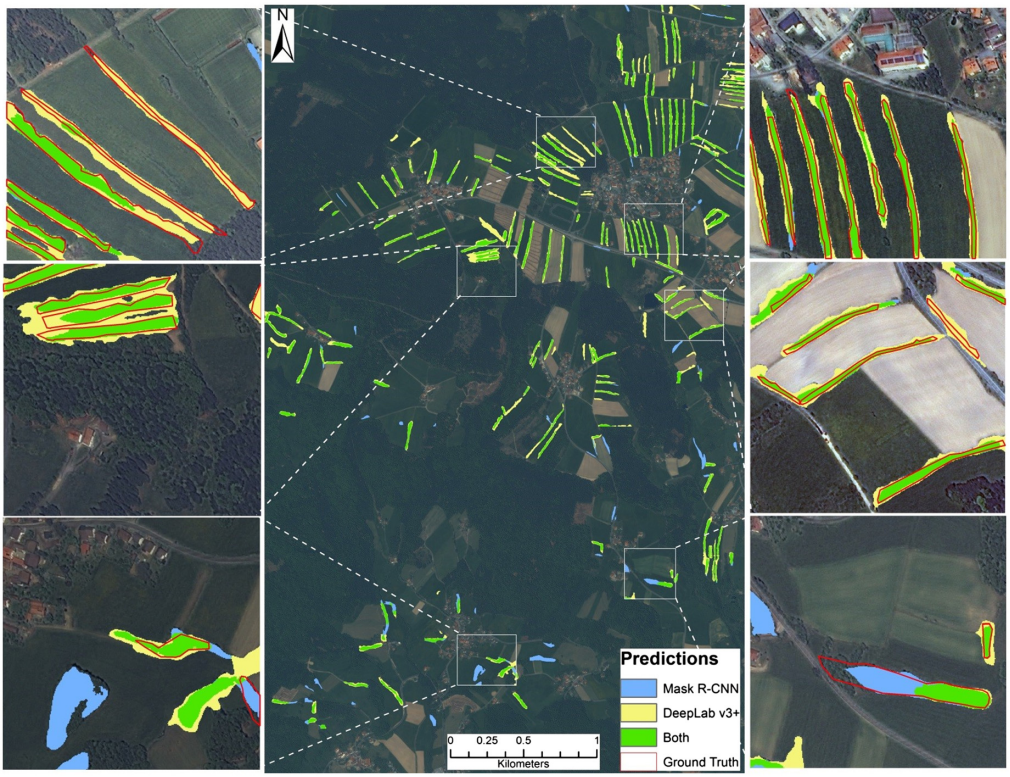

Fig. 8 Map showing hedgerow detections at a landscape scale within the study area. Zoomed in areas are of 320 × 320 pixel areas.

From a quantitative standpoint, Mask R-CNN outperformed DeepLab v3+ on per-pixel accuracy, whereas DeepLab v3+ had higher $F1$-score, precision, and recall across all detection thresholds (Table 3). Qualitatively, the predictions from Mask R-CNN performed better with regards to the accuracy of mask boundaries, as the boundaries of mask predictions made by DeepLab v3+ were overestimated in most cases (Fig. 8). In areas where only small gaps separated hedgerows, DeepLab v3+ failed to individually mask each hedgerow, instead adjacent hedgerows were predicted by one singular mask [Figs. 8(b)]. Mask R-CNN struggled the most with the identification of longer diagonal hedgerows, leading of only partial predictions, or complete omission of some hedgerows [Fig. 8(a)]. Predictions made by DeepLab v3+ were unaffected by angular orientation.

5.5 *Final Predictions Map*

Given that the DeepLab v3+ network trained using G produced the highest performance metric scores, this network was used to make predictions across the entire area of satellite scenes that were available for Freyung-Grafenau (Fig. 9). The network was able to make hedgerow predictions across a large area and for image scenes that had not been included in the training or validation dataset.

6 Discussion

6.1 *Optimal Three Band Combination*

The GRNIR band combination was found to be the optimal three band combination for detecting hedgerows. Other researchers have also shown that green, red, and NIR bands are the most correlated to vegetation parameters,[74,75] and useful for quick human identification of vegetation within a scene.[76] However, GRNIR was neither optimal across all performance metrics or thresholds of detection, and differences between different band combinations were relatively small. As such, the choice of band combination seems to be of minimal concern.

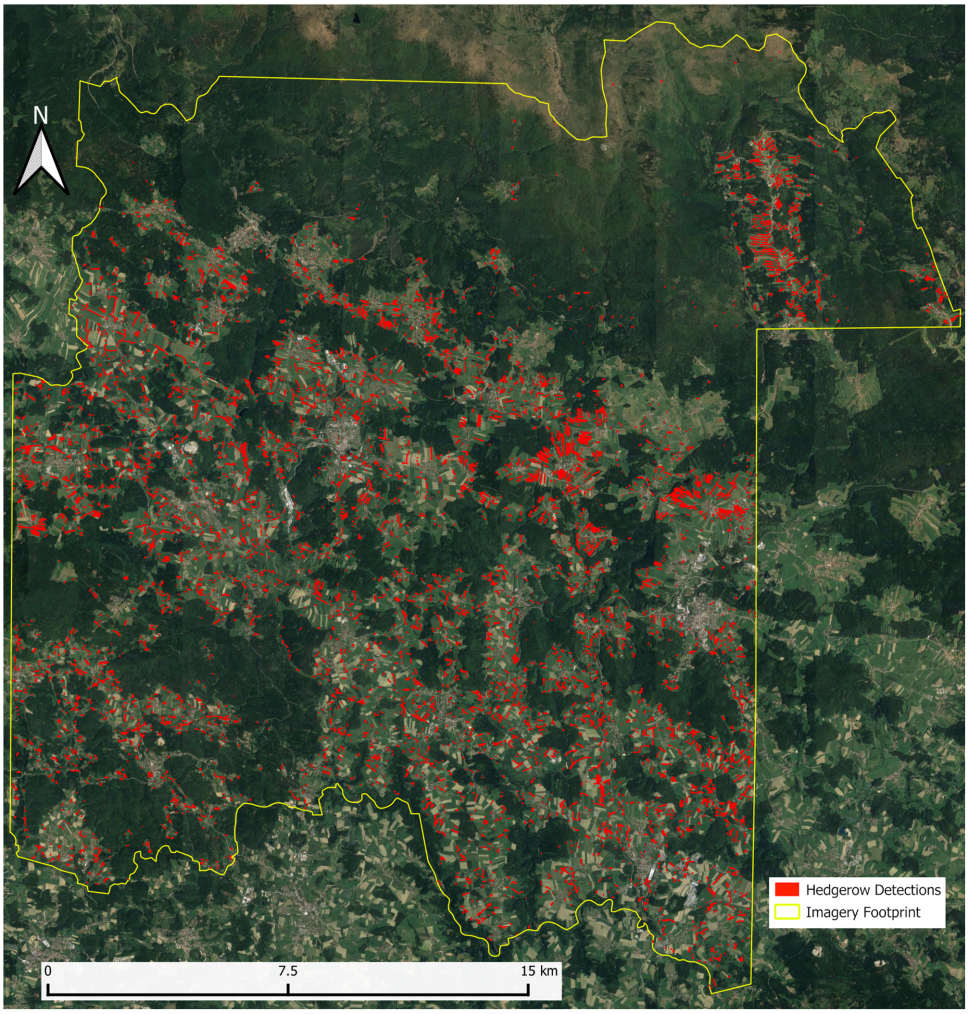

Fig. 9 Map showing the distribution of hedgerow predictions across the entire study area. Backdrop: Google™.

Previous works on hedgerow detection have all utilized the NIR band[20–26] and have demonstrated its importance when distinguishing vegetation types. However, we show here that RGB imagery performs at only slightly lower rates than band combinations that incorporated the NIR band. This performance is partially due to the network having been pretrained using RGB imagery but also suggests that NN are able to perform accurate classification using less spectral information due to the network's ability to extract morphological and contextual features. This shows that simple RGB cameras are sufficient for hedgerow monitoring when applying pretrained NNs. RGB cameras have a long history of use in aerial photography as they produce images which are easily interpretable at a relatively cheap cost.[77] Given that monitoring agencies often use RGB imagery for the manual digitization of hedgerows,[78] these flight campaigns could still be useful for hedgerow monitoring. Aerial imagery tends to be of higher spatial resolution[79] than satellite imagery, which may be of greater benefit than the addition of the NIR band, as research has shown that increased spatial resolution leads to improved results in CNNs.[27,80]

6.2 *Optimal Seasonal Imagery Input*

Images from October led to the best performance for hedgerow detection when dataset sizes were held constant. This is likely due to the unique spectral characteristics captured during this season, as hedgerows are typically composed of multiple bush and tree species, which vary in their phenological patterns.[81] Hedgerow images from the autumn season should thus contain more heterogeneous spectral features, which should help differentiating between spectrally similar

classes of forests and fields, which are typically more homogeneous, being dominated by one or a few species.[82,83]

Although images from October obtained the highest ratings across performance metrics, using the mixed seasons' dataset during training and October images for evaluation performed only a few percentages worse than using the October dataset in both stages. This supports the use of mixed images for network training, as predictions were not significantly harmed. Given that the comparison made here controlled for the size of datasets, the increased size garnered from using imagery from multiple seasons instead of a smaller single-season dataset should lead to performance increases that outweigh the decreased performance of using mixed rather than October images. This finding confirms the findings of others[40,67] that using mixed datasets lead to improved generalization when training neural networks.

The inclusion of imagery from varying phenological time periods can be conceptualized as a type of data augmentation, in that the network should have improved generalization by learning a variety of hedgerow spectral characteristics caused by different phenological stages, as well as differences in reflection intensities due to differences in sunlight incidence angles across seasons.[84] Thus, multi-season datasets may lead to more robust network predictions and may also improve predictions across gradients of elevation where phenological stages can differ over spatial scales of a few kilometers.[85,86] However, DeepLab v3+ was more sensitive to the use of mixed training images than Mask R-CNN, implying that DeepLab v3+ is less robust to spectral changes and augmentations than Mask R-CNN.

6.3 *Optimal Data Augmentation Strategy*

6.3.1 *Effects of individual augmentations*

When looking at geometric augmentations, both networks performed poorly when images were augmented through random flips in the image (vertical or horizontal). This could be because the flipping of hedgerows does not offer adequate variance to the dataset since hedgerow positions are simply mirrored. This may lead to bias toward hedgerows of a certain orientation and spectral features. Thus for the detection of relatively symmetrical objects the addition of flipped images may not be as beneficial as other augmentations. The results of the remaining three geometric augmentations (rotation of 45 deg, 90 deg, and scaling) were mostly dependent on the NN, as Mask R-CNN showed lower rates of $F1$-score, recall, and per-pixel accuracies, whereas DeepLab v3+ showed improvements when applying rotations. The use of rotations produces a greater variety of hedgerow orientations, providing the network more opportunity to learn a variety of orientation features. However, the negative effects seen in Mask R-CNN may stem from the network continually learning the same spectral features, leading to spectral bias.

Overall, it seems that the spectral augmentations helped to increase the number of TP while simultaneously reducing the number of FN in predictions made by Mask R-CNN. The addition of image noise led to the greatest improvements. Previous research has found the addition of noise to have negative effects on network performances.[68] Here however, it seems that the opposite is the case, as the addition of noise should force the network to learn to detect hedgerows using non-spectral features (e.g., orientation and contextual). Other spectral augmentations (add and blur) showed relatively unchanged $F1$-score, yet an increase in per-pixel accuracy. This suggests that spectral augmentations lead to more accurate mask boundaries by forcing the network to learn contextual and morphological features, rather than relying on spectral features, which are often similar to surrounding fields and forests.

Although Mask R-CNN obtained improved results when applying spectral augmentations, DeepLab v3+ performed poorly when trained with spectrally augmented images, as $F1$-scores decreased in all cases. The largest decrease in $F1$-score came from the contrast and contrast per-channel augmentations. Given that these two augmentations caused the largest changes to pixel values implies that DeepLab v3+ is sensitive to large spectral changes during augmentation.

In cases where hedgerow predictions were TP, both networks produced more accurate mask boundaries using spectrally augmented training images. This effect of increased boundary accuracy when using spectral augmentations has been found by Stiller et al.,[67] where the application of a hue transformation increased the accuracy of urban building mask boundaries.

However, the use of spectrally augmented images did introduce more FP detections, especially when using DeepLab v3+. Ma et al.[40] found the opposite effect, as DeepLab v3+ achieved greater TP object detections when augmenting the spectral values of RGB images for goat detection. This conflicting result suggests that the application of spectral augmentations may be better suited to detection tasks where spectral characteristics are less of a defining feature of the target objects (e.g., buildings). This is sensible from a remote sensing perspective as spectral augmentations may destroy the characteristic spectral profile that differing vegetative surfaces possess, leading to misclassifications of hedgerow as forest.

6.3.2 *Effects of applying multiple augmentations*

The effect of applying all augmentations was investigated by training the networks on two different datasets (*G* and GS). Overall, the increased size of these two datasets compared to the original unaugmented dataset led to large increases in performances for both Mask R-CNN and DeepLab v3+.

Using Mask R-CNN, our results show that the addition of spectral augmentations helped make the network more generalizable, leading to the acceptance of more hedgerow pixels into mask predictions, thus masking the full extent of hedgerows. However, given that recall rates were similar at lower detection thresholds implies that the use of spectral augmentations did not lead to a large increase in overall hedgerow detections, but rather that hedgerows detected only partially by the network trained with *G* were masked more fully using GS. The higher precision from GS decreased at the lower thresholds, leading to more precise predictions using *G*. This is because while more hedgerows were detected using GS, false detections also increased. Given the trade-off between recall and precision, the decision to incorporate spectral augmentations into the training data depends on the desired outcome of predictions.

With respect to DeepLab v3+, experiments using *G* and GS reinforced the conclusions from the individual augmentation experiments. The inclusion of spectral augmentations from GS led to decreases in model performance compared to training with *G*. Although the performance metrics show only a few percentages of difference between the two, visual inspection of predictions showed the problem of using spectral augmentations, as some large forested areas were classified as hedgerows. Objects detected as FP from the other trained networks typically shared some contextual or morphological characteristics with hedgerows. However, the addition of spectral augmentations in DeepLab v3+ led to FP, which possessed few hedgerow characteristics, as these predictions did not neighbor open fields and were devoid of linear shape. This goes against the notion that spectral augmentations lead to improved learning of morphological features as suggested in past works.[40,67] These types of predictions appear similar to results obtained in the presence of adversarial examples. Adversarial examples typically involve the addition of relatively imperceivable spectral alterations, such as noise, resulting in highly inaccurate network predictions.[87] As such, it is possible that the inclusion of spectral augmentations into the training data caused DeepLab v3+ to produce these FP detections, which lacked any semblance to hedgerows.

6.4 *Overall Performances of Mask R-CNN and DeepLab v3+*

FPs typically occurred with objects that resemble hedgerows or vegetated areas directly neighboring fields [Fig. 6(g) and 6(i)]. Research has found that species compositions tend to be similar between hedgerows and forest edge areas,[82,88] which may lead to higher rates of misclassifications of forest edges as hedgerows. LfU defined hedgerows as linear structures comprised of a mixture of bush and tree species, being a minimum of two years old. Thus, numerous linear vegetation structures that appear similar to hedgerows from a satellite (e.g., tree lines and young hedgerows) were often "misclassified" as hedgerows, leading to reduced precision rates. However, such features would likely be just as difficult for a human observer to correctly label given the same image, and thus the only way to improve such misclassifications would be through field surveys or the addition of auxiliary data (e.g., LiDAR).

The per-pixel accuracy metric was partially affected by the ground truth polygons occasionally being thinner than the hedgerow itself, leading to correct pixel classifications being penalized, introducing a negative bias.

This study found that DeepLab v3+ outperformed Mask R-CNN in $F1$-score, precision, and recall measures. To date only one other study has performed instance/semantic segmentation for vegetation objects.[44] In their study, Zhao et al.[44] compared Mask R-CNN to an FCN network known as U-Net[45] for detection of pomegranate trees. Their results achieved much higher rates of precision and recall for Mask R-CNN than achieved in this paper. However, this was likely due to the simplified detection task of Zhao et al.,[44] as training and validation images were comprised of the target trees against a bare soil background. In our study, hedgerow objects were typically located within spectrally similar areas (e.g., pasture fields and forest edges), making the separation of hedgerows from the background more difficult. Additionally, images used for detecting hedgerows often contained instances of wooded vegetation, which the network could potentially confuse as hedgerows. Zhao et al.[44] found that Mask R-CNN outperformed U-Net on both precision and recall, whereas we found DeepLab v3+ performed better than Mask R-CNN. This is likely due to U-Net being an older FCN architecture, which lacks the use of atrous convolutions. This suggests the importance of the use of atrous convolutions within NN architectures used for remote sensing image analysis, as they provide an increased receptive field without downsampling the spatial resolution of the input.[59]

Zhao et al.[44] also found the predicted masks of Mask R-CNN to be more accurate, as the FCN mask predictions would span over multiple individual objects in areas where objects were densely clustered. This same result was also found here in the case of hedgerows, as DeepLab v3+ also had troubles with separating object masks when individual objects were close together. This is likely due to the use of the FPN in Mask R-CNN, which provides two main benefits. First, it performs upsampling at a slower rate (rate of 2) compared to DeepLab v3+ (rate of 4). Slower upsampling rates have been found to lead to better mask outputs[37] and allow for more residual connections during upsampling steps. Second, it allows for fine-scale objects to be masked using fine-scale feature maps. Thus objects are masked using more appropriately scaled feature layers, which has been shown to benefit object classifications.[89,90]

An inability to separate hedgerows with individual masks can have negative effects on monitoring programs, as the number of hedgerows would be underestimated. Additionally, the loss of hedgerows in areas of dense occurrences may not be detected, as a mask containing multiple hedgerows may only decrease in size. Given that hedgerow structural features, such as width, influence mammal and avifaunal species abundance, and richness,[12] the accurate delineation of hedgerow masks would be important if used for certain environmental models and monitoring.

Mask R-CNN struggled to detect and fully mask long diagonal hedgerows. There appear to be two main issues. The first is caused by the interaction between the RPN and FPN of Mask R-CNN. Diagonal hedgerows require larger anchors to encapsulate them (Fig. 10). Given that the RPN of Mask R-CNN classifies larger anchors using layers of the FPN characterized by coarse spatial resolution, diagonal hedgerows are detected using coarse feature maps. Since hedgerows are fine-scale objects, they require fine-scale spatial resolution feature maps to be accurately classified and masked,[89,90] such as those at the lower levels of the FPN. The second issue with detecting long diagonal hedgerows is the large class imbalance within anchors of diagonal hedgerows as much of the area is dominated by background pixels.

A solution to both issues would be the use of rotational anchors, allowing for anchor bounding boxes with dimensions that share a relationship with the physical size of the object.[72] Rotated anchors would thus reduce RoI scales, in turn mapping objects to more appropriate layers within the FPN, as well as reducing the amount of background pixels within the bounding box. Rotational anchors could also improve hedgerow detections within areas of dense hedgerow occurrences, as Mask R-CNN also struggled in such areas. This is because Mask R-CNN assumes an anchor contains a singular instance of a target object.[91] However, in cases of densely occurring objects, anchors are unable to separate individual objects. This is most often the case when dealing with diagonal hedgerows, as their large bounding boxes often contain neighboring objects (Fig. 10).

6.5 *Comparison with Object-Based Studies*

Overall, our results were superior to those of existing OBIA approaches for hedgerow detection on either one[21,23] or both[22,25] of precision and recall. Our approach is more straightforward compared to existing OBIA approaches, as they require the engineering and testing of many feature

Fig. 10 Anchor bounding box without (red) and with (yellow) rotation for a single hedgerow.

variables, whereas the neural networks required only the raw (and augmented) imagery and training labels. Given that past OBIA studies have come to different conclusions regarding the optimal features for hedgerow classification, new applications of such methods would also require testing of different feature variables. This also suggests that features selected for one study area will generalize poorly across larger spatial scales. Past OBIA studies have utilized higher resolution imagery (0.5 to 0.7 m) than this current study (1 m). We thus show a variation from previous works by demonstrating that coarser resolution imagery can be applied to hedgerow monitoring tasks, thus extending the base of suitable remote sensing data and potentially decreasing acquisition costs. Given that higher resolution has been shown to improve NN performance,[27,80] it is likely that our results would have outperformed OBIA by a greater margin given higher image resolutions. Finally, past OBIA studies were performed on much smaller areas (5 to 12 km^2), whereas this study produced results across a 562 km^2 area. These findings show that pretrained NN offer improvement in the performance and practicality over OBIA with respect to hedgerow mapping, thus making them a useful tool for monitoring agencies.

7 Conclusions

This work successfully produced a hedgerow detection map across a large spatial scale using a practical methodology. Previously established object-based detection methods require expert knowledge of feature engineering and remote sensing which pose as barriers to the implementation of such an approach. Additionally, such methods are unsuitable for detections across large regional scales given the reliance on complex rule sets and feature engineering. The pretrained NNs investigated here offer a practical method for hedgerow detection by eliminating the need for manual feature engineering and selection involved in OBIA. In addition to the practicality of the method, results of pretrained NNs also outperformed those of OBIA approaches, supporting the use of pretrained NN for large scale landscape mapping tasks. Although the use of custom designed NN trained from scratch may provide better results, such approaches come at the expense of much greater efforts and lower accessibility due to the technical requirements.

Overall, both pretrained DeepLab v3+ and Mask R-CNN networks were capable of hedgerow mapping across a large spatial scale (562 km^2), thus demonstrating the spatial scale at which neural networks are capable of making hedgerow detections. Both networks were trained using minimal input, as only the ground truth annotations and matching satellite images were used. Mask R-CNN was able to produce mask predictions with more accurate object mask boundaries (40.6%) than DeepLab v3+ (36.9%), likely due to the inclusion of the FPN. However, DeepLab v3+ greatly outperformed Mask R-CNN in $F1$-score (74.7%), precision (69.4%), and recall (80.8%).

Atrous convolutions used in DeepLab v3+ are seemingly important architectural features for NNs designed for remote sensing tasks as they eliminate downsampling steps while

simultaneously increasing the receptive field of the network. The main limitation with Mask R-CNN was the poor detection of long diagonally oriented hedgerows. We postulate this is due to the current rigid anchor implementation. Thus any future network designs incorporating anchors to detect hedgerows, or other objects which are characterized by uneven aspect ratios, should incorporate anchors with trainable rotation parameters.

Testing different three band combinations found that GRNIR was the optimal choice for hedgerow detection. Given that other band combinations performed similarly well, the choice of band combination does not play as large a role. This finding goes against previous research where the NIR band has been found to be a highly significant feature for hedgerow detection. As such, the need for expensive multi-spectral imagery could be replaced with RGB cameras, allowing for a more affordable approach. Datasets using images from mixed seasons should be preferred over a single-season dataset as this both increases the dataset size and should increase model generalizability, as inclusion of differing seasonal imagery acts as a form of data augmentation by exposing the network to seasonal variability of spectral features for the target class. Mask R-CNN was found to be more robust than DeepLab v3+ regarding the inclusion of multi-seasonal imagery, as well as toward all other forms of spectral augmentations applied. Thus, spectral augmentations should be applied with caution as the efficacy appears to be network dependent, whereas geometric augmentations can be applied regardless of network. Although this work focuses only on hedgerows, our results regarding data augmentation techniques should be transferable to other vegetation landscape features.

Pretrained networks offer practical solutions to remote sensing tasks. Currently, pretrained weights for networks trained on large image datasets such as COCO are readily available online, but there are few open source weights from networks trained on large remote sensing datasets. Open access to such weights would be beneficial to the conservation community and further facilitate and improve the use of pretrained NN for remote sensing in cases where the target dataset size is limited.

8 Appendix A

Appendix A contains supplementary Figs. 11–15 to the text.

Fig. 11 Single 320×320 m image tile showing an example of some roughly digitized hedgerow annotations (red) created and provided by the Bavarian Landesamt fuer Umwelt.

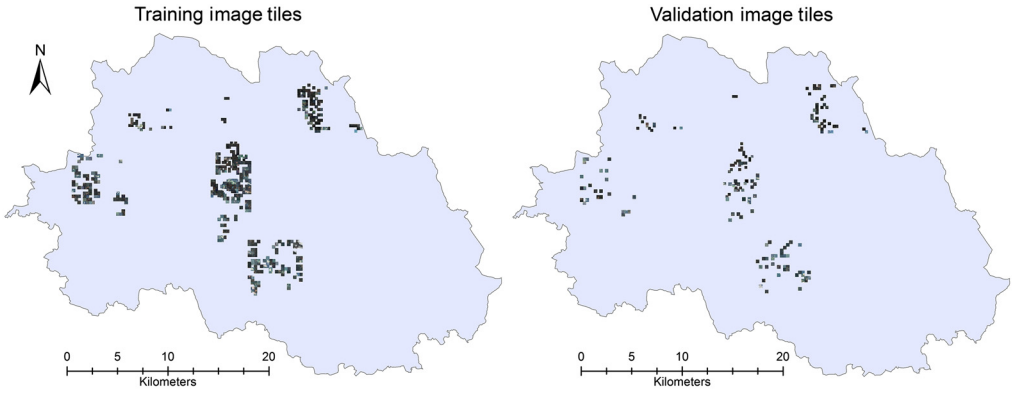

Fig. 12 Distribution of both training and validation image tiles across the study area.

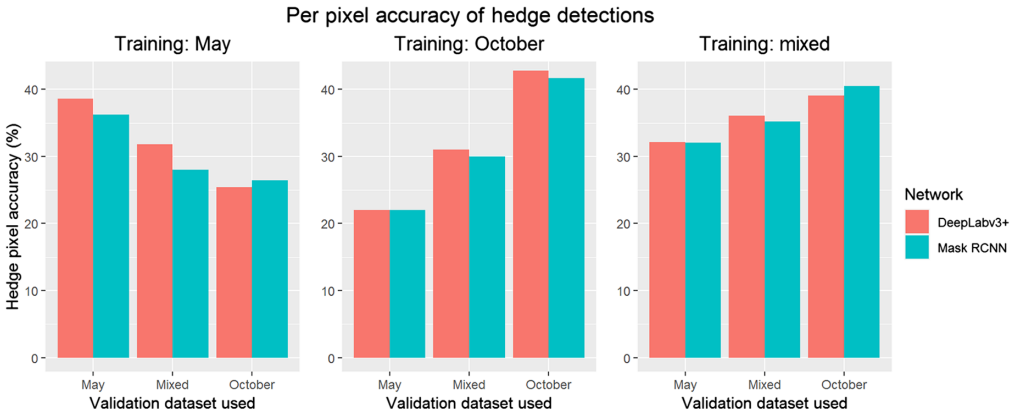

Fig. 13 Per-pixel accuracy scores for both DeepLab and Mask R-CNN using three different seasonal datasets (May, October, and a mixture of the two) split into training and validation sets.

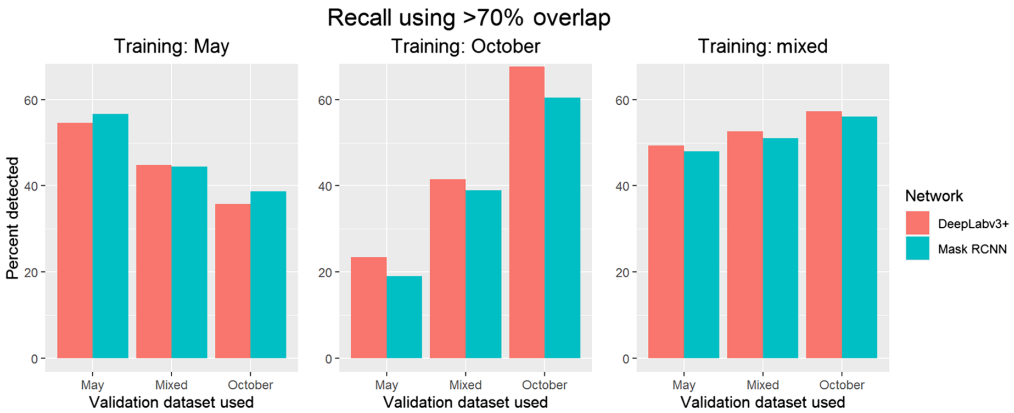

Fig. 14 Recall scores for both DeepLab and Mask R-CNN using three different seasonal datasets (May, October, and a mixture of the two) split into training and validation sets.

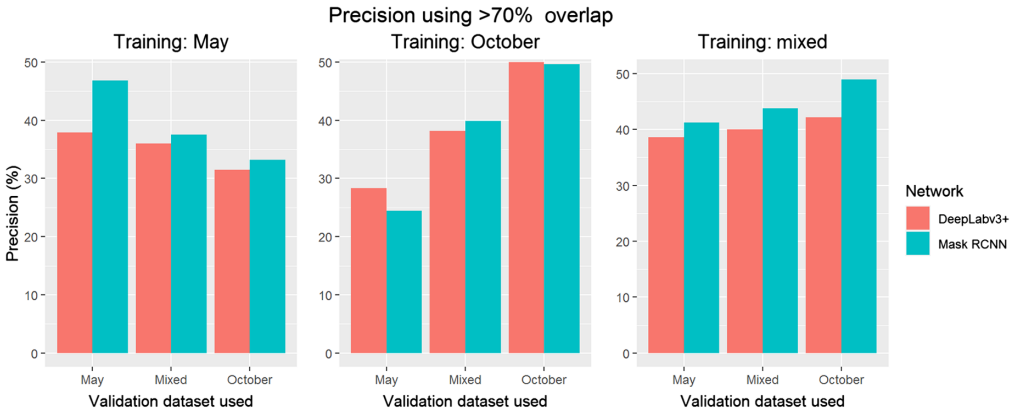

Fig. 15 Precision scores for both DeepLab v3+ and Mask R-CNN using three different seasonal datasets (May, October, and a mixture of the two) split into training and validation sets.

Acknowledgments

The authors acknowledge the support from Bayerisches Landesamt fuer Umwelt in providing hedgerow training data, as well as supporting us with information regarding the collection protocol for the training data. Includes material © 2003, EUSI GmbH, all rights reserved. The authors declare no conflicts of interest.

References

1. S. Endenburg et al., "The homogenizing influence of agriculture on forest bird communities at landscape scales," *Landsc. Ecol.* **34**, 2385–2399 (2019).
2. F. Sánchez-Bayo and K. A. G. Wyckhuys, "Worldwide decline of the entomofauna: a review of its drivers," *Biol. Conserv.* **232**, 8–27 (2019).
3. N. Dudley and S. Alexander, "Agriculture and biodiversity: a review," *Biodiversity* **18**, 45–49 (2017).
4. A. Chaudhary and T. Kastner, "Land use biodiversity impacts embodied in international food trade," *Glob. Environ. Change* **38**, 195–204 (2016).
5. M. S. Meier et al., "Environmental impacts of organic and conventional agricultural products—are the differences captured by life cycle assessment?" *J. Environ. Manage.* **149**, 193–208 (2015).
6. J. Ekroos, J. Heliölä, and M. Kuussaari, "Homogenization of lepidopteran communities in intensively cultivated agricultural landscapes," *J. Appl. Ecol.* **47**, 459–467 (2010).
7. A. G. Power, "Ecosystem services and agriculture: tradeoffs and synergies," *Philos. Trans. R. Soc.y B: Biol. Sci.* **365**, 2959–2971 (2010).
8. D. Tilman et al., "Forecasting agriculturally driven global environmental change," *Science* **292**, 281–284 (2001).
9. S. Petit et al., "Field boundaries in Great Britain: stock and change between 1984, 1990 and 1998," *J. Environ. Manage.* **67**, 229–238 (2003).
10. J. Baudry et al., "Hedgerows: an international perspective on their origin, function and management," *J. Environ. Manage.* **60**, 7–22 (2000).
11. C. Vannier and L. Hubert-Moy, "Multiscale comparison of remote-sensing data for linear woody vegetation mapping," *Int. J. Remote Sens.* **35**, 7376–7399 (2014).
12. E. Padoa-Schioppa et al., "Bird communities as bioindicators: the focal species concept in agricultural landscapes," *Ecol. Indic.* **6**, 83–93 (2006).
13. A. Lacoeuilhe et al., "The relative effects of local and landscape characteristics of hedgerows on bats," *Diversity* **10**, 72 (2018).
14. J. T. Staley et al., "Long-term effects of hedgerow management policies on resource provision for wildlife," *Biol. Conserv.* **145**, 24–29 (2012).
15. J. S. P. Froidevaux, M. Broyles, and G. Jones, "Moth responses to sympathetic hedgerow management in temperate farmland," *Agric. Ecosyst. Environ.* **270**, 55–64 (2019).

16. L. A. Morandin, R. F. Long, and C. Kremen, "Hedgerows enhance beneficial insects on adjacent tomato fields in an intensive agricultural landscape," *Agric. Ecosyst. Environ.* **189**, 164–170 (2014).
17. P. W. Bright, "Behaviour of specialist species in habitat corridors: arboreal dormice avoid corridor gaps," *Anim. Behav.* **56**, 1485–1490 (1998).
18. J. Holden et al., "The role of hedgerows in soil functioning within agricultural landscapes," *Agric. Ecosyst. Environ.* **273**, 1–12 (2019).
19. A. Lotfi et al., "Interdisciplinary analysis of hedgerow network landscapes' sustainability," *Landsc. Res.* **35**, 415–426 (2010).
20. K. Tansey et al., "Object-oriented classification of very high resolution airborne imagery for the extraction of hedgerows and field margin cover in agricultural areas," *Appl. Geogr.* **29**, 145–157 (2009).
21. M. Bock et al., "Object-oriented methods for habitat mapping at multiple scales—case studies from Northern Germany and Wye Downs, UK," *J. Nat. Conserv.* **13**, 75–89 (2005).
22. S. Aksoy, H. G. Akçay, and T. Wassenaar, "Automatic mapping of linear woody vegetation features in agricultural landscapes using very high resolution imagery," *IEEE Trans. Geosci. Remote Sens.* **48**, 511–522 (2009).
23. J. O'Connell, U. Bradter, and T. G. Benton, "Wide-area mapping of small-scale features in agricultural landscapes using airborne remote sensing," *ISPRS J. Photogramm. Remote Sens.* **109**, 165–177 (2015).
24. C. Vannier and L. Hubert-Moy, "Detection of wooded hedgerows in high resolution satellite images using an object-oriented method," in *Proc. Int. Geosci. and Remote Sens. Symp.*, Boston, Massachusetts (2008).
25. D. Ducrot, A. Masse, and A. Ncibi, "Hedgerow detection in HRS and VHRS images from different source (optical, radar)," in *Proc. Int. Geosci. and Remote Sens. Symp.*, Munich (2012).
26. M. Fauvel et al., "Hedges detection using local directional features and support vector data description," in *Proc. Int. Geosci. and Remote Sens. Symp.*, Munich (2012).
27. M. Wurm et al., "Semantic segmentation of slums in satellite images using transfer learning on fully convolutional neural networks," *ISPRS J. Photogramm. Remote Sens.* **150**, 59–69 (2019).
28. Y. Lecun et al., "Gradient-based learning applied to document recognition," *Proc. IEEE* **86**, 2278–2324 (1998).
29. W. Li et al., "Large-scale oil palm tree detection from high-resolution satellite images using two-stage convolutional neural networks," *Remote Sens.* **11**, 11 (2019).
30. T. Liu et al., "Comparing fully convolutional networks, random forest, support vector machine, and patch-based deep convolutional neural networks for object-based wetland mapping using images from small unmanned aircraft system," *GISci. Remote Sens.* **55**, 243–264 (2018).
31. N. Kussul et al., "Deep learning classification of land cover and crop types using remote sensing data," *IEEE Geosci. Remote Sens. Lett.* **15**, 778–782 (2017).
32. M. Längkvist et al., "Classification and segmentation of satellite orthoimagery using convolutional neural networks," *Remote Sens.* **8**, 329 (2016).
33. T. Ishii et al., "Surface object recognition with CNN and SVM in Landsat 8 images," in *Proc. 14th IAPR Int. Conf. Mach. Vision Appl.*, Tokyo (2015).
34. P. M. Atkinson and A. R. L. Tatnall, "Introduction neural networks in remote sensing," *Int. J. Remote Sens.* **18**, 699–709 (1997).
35. K. He et al., "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vision*, Venice (2017).
36. T. Panboonyuen et al., "Road segmentation of remotely-sensed images using deep convolutional neural networks with landscape metrics and conditional random fields," *Remote Sens.* **9**, 680 (2017).
37. J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, Boston, Massachusetts (2015).

38. R. Girshick et al., "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognit.*, Columbus, Ohio (2014).
39. L.-C. Chen et al., "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Comput. Vision—ECCV 2018, Proc. Eur. Conf. Comput. Vision*, Munich (2018).
40. R. Ma, P. Tao, and H. Tang, "Optimizing data augmentation for semantic segmentation on small-scale dataset," in *Proc. 2nd Int. Conf. Control and Comput. Vision*, Jeju (2019).
41. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM* **60**, 84–90 (2012).
42. A. Fawzi et al., "Adaptive data augmentation for image classification," in *Proc. Int. Conf. Image Process.*, Pheonix, Arizona (2016).
43. N. Dvornik, J. Mairal, and C. Schmid, "On the importance of visual context for data augmentation in scene understanding," *IEEE Trans. Pattern Anal. Mach. Intell.*, 1–15 (2019).
44. T. Zhao et al., "Comparing U-Net convolutional network with Mask R-CNN in the performances of pomegranate tree canopy segmentation," *Proc. SPIE* **10780**, 107801J (2018).
45. O. Ronneberger, P. Fischer, and T. Brox, "U-Net: convolutional networks for biomedical image segmentation," in *Proc. 18th Int. Conf. Med. Image Comput. and Comput.-Assist. Interv.*, Munich (2015).
46. J. F. Mullen, F. R. Tanner, and P. A. Sallee, "Comparing the effects of annotation type on machine learning detection performance," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR) Workshops*, Long Beach (2019).
47. T. Y. Lin et al., "Microsoft COCO: common objects in context," in *Comput. Vision—ECCV 2014, Proc. 13th Eur. Conf. Comput. Vision*, Zurich (2014).
48. U. Bernhards et al., "Pilotfallstudie zur Bewertung der Ausgleichszulage in benachteiligten Gebieten im Landkreis Freyung-Grafenau," Arbeitsbericht//Bundesforschungsanstalt für Landwirtschaft (FAL), Institut für Betriebswirtschaft, Agrarstruktur und Ländliche Räume (2003).
49. D. Palandro et al., "Change detection in coral reef communities using Ikonos satellite sensor imagery and historic aerial photographs," *Int. J. Remote Sens.* **24**, 873–878 (2003).
50. A. Laben and B. V. Brower, "Process for enhancing the spatial resolution of multispectral imagery using pansharpening," Google Patents US006011875A (2000).
51. D. Marmanis et al., "Classification with an edge: improving semantic image segmentation with boundary detection," *ISPRS J. Photogramm. Remote Sens.* **135**, 158–172 (2018).
52. R. Girshick et al., "Region-based convolutional networks for accurate object detection and segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.* **38**, 142–158 (2016).
53. R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vision*, Santiago, Chile (2015).
54. S. Ren et al., "Faster R-CNN: towards real-time object detection with region proposal networks," in *IEEE Trans. Pattern Anal. Mach. Intell.* **39**(6), 1137–1149 (2015).
55. T.-Y. Lin et al., "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, Honolulu, Hawaii (2017).
56. K. He et al., "Deep residual learning for image recognition," in *Proc. 2016 IEEE Conf. Comput. Vision and Pattern Recognit.*, Las Vegas, Nevada (2016).
57. J. Yosinski et al., "How transferable are features in deep neural networks," in *NIPS'14, Proc. 27th Int. Conf. Neural Inf. Process. Syst.*, Montreal, Quebec, MIT Press, Cambridge, Massachusetts (2014).
58. F. Chollet, "Xception: deep learning with depthwise separable convolutions," in *Proc. 30th IEEE Conf. Comput. Vision and Pattern Recognit.*, Honolulu, Hawaii (2017).
59. L.-C. Chen et al., "Rethinking atrous convolution for semantic image segmentation," arXiv:1706.05587, pp. 1–14 (2017).
60. M. Mel, U. Michieli, and P. Zanuttigh, "Incremental and multi-task learning strategies for coarse-to-fine semantic segmentation," *Technologies* **8**, 1 (2020).
61. T.-Y. Lin et al., "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vision*, Venice (2017).

62. Y. Yao et al., "Ship detection in optical remote sensing images based on deep convolutional neural networks," *J. Appl. Remote Sens.* **11**, 042611 (2017).
63. M. Xie et al., "Transfer learning from deep features for remote sensing and poverty mapping," in *Proc. 30th AAAI Conf. Artif. Intell.*, Pheonix, Arizona (2016).
64. O. A. B. Penatti, K. Nogueira, and J. A. D. Santos, "Do deep features generalize from everyday objects to remote sensing and aerial scenes domains?," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognit. Workshops*, Boston, Massachusetts (2015).
65. L. Ma et al., "Deep learning in remote sensing applications: a meta-analysis and review," *ISPRS J. Photogramm. Remote Sens.* **152**, 166–177 (2019).
66. A. Voulodimos et al., "Deep learning for computer vision: a brief review," *Comput. Intell. Neurosci.* **2018**, 1–13 (2018).
67. D. Stiller et al., "Large-scale building extraction in very high-resolution aerial imagery using Mask R-CNN," in *Proc. 2019 Joint Urban Remote Sens. Event*, Vannes (2019).
68. S. Dodge and L. Karam, "Understanding how image quality affects deep neural networks," in *Proc. 18th Int. Conf. Qual. Multimedia Exp.*, Lisbon (2016).
69. X.-Y. Tong et al., "Land-cover classification with high-resolution remote sensing images using transferable deep models," *Remote Sens. Environ.*, **237**, 111322 (2020).
70. G. Mountrakis, J. Im, and C. Ogole, "Support vector machines in remote sensing: a review," *ISPRS J. Photogramm. Remote Sens.* **66**, 247–259 (2011).
71. G. Csurka, D. Larlus, and F. Perronnin, "What is a good evaluation measure for semantic segmentation?," in *Proc. Br. Mach. Vision Conf.*, Cardiff, Wales (2013).
72. Q. Wen et al., "Automatic building extraction from Google Earth images under complex backgrounds based on deep instance segmentation network," *Sensors* **19**, 333 (2019).
73. K. Zhao et al., "Building extraction from satellite images using mask R-CNN with building boundary regularization," in *Proc. IEEE/CVF Conf. Comput. Vision and Pattern Recognit. Workshops*, Salt Lake City, Utah (2018).
74. A. B. Baloloy et al., "Estimation of mangrove forest aboveground biomass using multispectral bands, vegetation indices and biophysical variables derived from optical satellite imageries: rapideye, planetscope, and Sentinel-2," *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.* **IV-3** 29–36 (2018).
75. K. P. Price, X. Guo, and J. M. Stiles, "Optimal landsat TM band combinations and vegetation indices for discrimination of six grassland types in eastern Kansas," *Int. J. Remote Sens.* **23**, 5031–5042 (2002).
76. P. Baecka et al., "High resolution vegetation mapping with a novel compact hyperspectral camera system," in *Proc. 13th Int. Conf. Precis. Agric.*, St. Louis, Missouri (2016).
77. M. Pepe, L. Fregonese, and M. Scaioni, "Planning airborne photogrammetry and remote-sensing missions with modern platforms and sensors," *Eur. J. Remote Sens.* **51**, 412–436 (2018).
78. J. W. Dover, *The Ecology of Hedgerows and Field Margins*, 1st ed., Abingdon-on-Thames, Routledge (2019).
79. M. Ruwaimana et al., "The advantages of using drones over space-borne imagery in the mapping of mangrove forests," *PLoS One* **13**, e0200288 (2018).
80. A. Farooq, J. Hu, and X. Jia, "Analysis of spectral bands and spatial resolutions for weed classification via deep convolutional neural network," *IEEE Geosci. Remote Sens. Lett.* **16**, 183–187 (2018).
81. R. T. T. Forman and J. Baudry, "Hedgerows and hedgerow networks in landscape ecology," *Environ. Manage.* **8**, 495–510 (1984).
82. A. Ghiyamat and H. Z. M. Shafri, "A review on hyperspectral remote sensing for homogeneous and heterogeneous forest biodiversity assessment," *Int. J. Remote Sens.* **31**, 1837–1856 (2010).
83. T. Hycza, K. Stereńczak, and R. Bałazy, "Potential use of hyperspectral data to classify forest tree species," *N. Z. J. For. Sci.* **48**, 1–13 (2018).
84. S. Liu et al., "ERN: edge loss reinforced semantic segmentation network for remote sensing images," *Remote Sens.* **10**, 1339 (2018).

85. Y. Vitasse et al., "Altitudinal differentiation in growth and phenology among populations of temperate-zone tree species growing in a common garden," *Can. J. For. Res.* **39**, 1259–1269 (2009).
86. C. Ziello et al., "Influence of altitude on phenology of selected plant species in the Alpine region (1971–2000)," *Clim. Res.* **39**, 227–234 (2009).
87. C. Xie et al., "Adversarial examples for semantic segmentation and object detection," in *Proc. IEEE Int. Conf. Comput. Vision (ICCV)*, Venice (2017).
88. L. S. Herlin and G. L. A. Fry, "Dispersal of woody plants in forest edges and hedgerows in a Southern Swedish agricultural area: the role of site and landscape structure," *Landsc. Ecol.* **15**, 229–242 (2000).
89. K. Zhou et al., "Building segmentation from airborne VHR images using Mask R-CNN," *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.—ISPRS Arch.* **XLII-2/W13**, 155–161 (2019).
90. T. Blaschke, "Object based image analysis for remote sensing," *ISPRS J. Photogramm. Remote Sens.* **65**, 2–16 (2010).
91. L. Liu, Z. Pan, and B. Lei, "Learning a rotation invariant detector with rotatable bounding box," arXiv:1711.09405, pp. 1–9 (2017).

Steve Ahlswede received his BSc degree in environment and resource management from Brandenburg University of Technology, Cottbus, Germany, and his Msc degree in environmental sciences from the University of Trier in 2020. He is a research associate at the University of Trier. His research interests are in precision agriculture and applications of machine learning for image processing.

Sarah Asam received her master's degree in global change ecology from the University of Bayreuth in 2010 and her PhD in remote sensing from the University of Würzburg in 2014. She is a research associate at the German Remote Sensing Data Center of the German Aerospace Center. Her research interests include remote sensing of vegetation, monitoring of agroecosystems, radiation transfer modeling, and time series analysis.

Achim Röder received his PhD in natural sciences from Trier University in 2005. He is an academic in the Department of Environmental Remote Sensing and Geoinformatics at Trier University. His main scientific interest is on the integration of optical remote sensing and geospatial data analysis for environmental studies with a particular focus on the assessment of temporal dynamics of land use/cover.